Project Ninja Run

1. Problem Statement : Ninja Run Game

Ninja Run game is a game which ninja run and avoid many obstruct that appear on the screen. How long ninja can run , the score will increase. Ninja can take damage in 5 times. So ninja must collect items , Game has two types of items. The first is sushi can heal, Second is scroll which collect in 3 times ninja will can use ultimate skill.

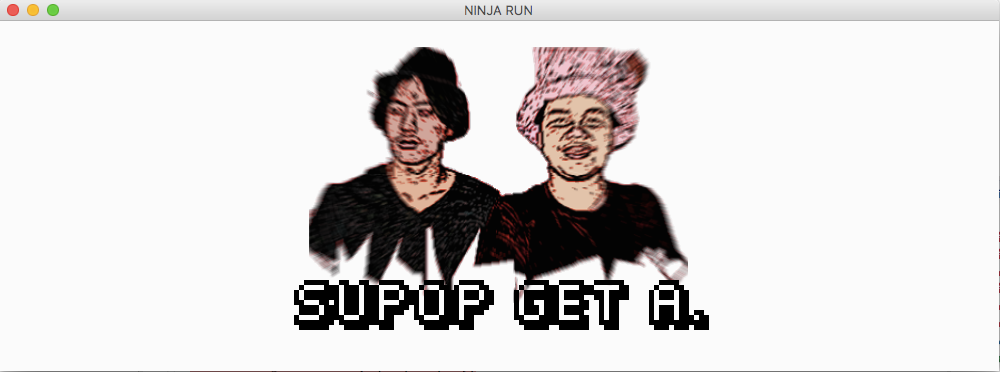
The game has 6 states. The intro state is the first one shown up when player open game to present my team.

Figure 1 : The intro state of the application.

At intro state player can press ENTER to skip / wait a little time to switch to menu state.

The menu state is the second one to let player select option .

Figure 2: the menu state of application.

At this menu state , player can .

* Press Enter to select option.
* Up/Down arrow to move option.

When player press Enter on the Start option. The program’s window switches to the playing state

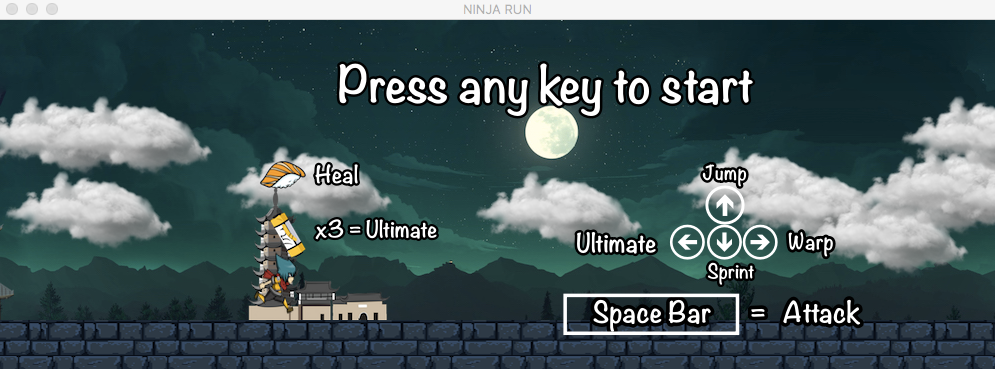
The playing state has 4 sub state. The start game state is the first one to show how to control and what items that can collect them.

Figure 3: the start game state of playing state .

At start game state , player can.

* Press any keys to start game.

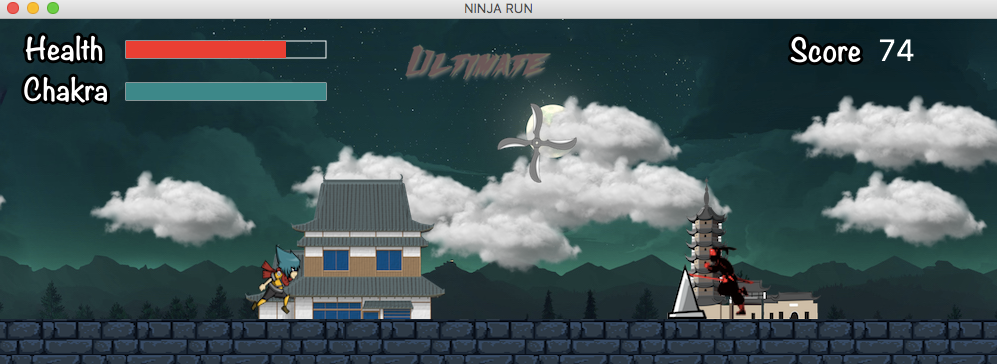
When player press any key the game will start.

Figure 3 :the game play state of playing state.

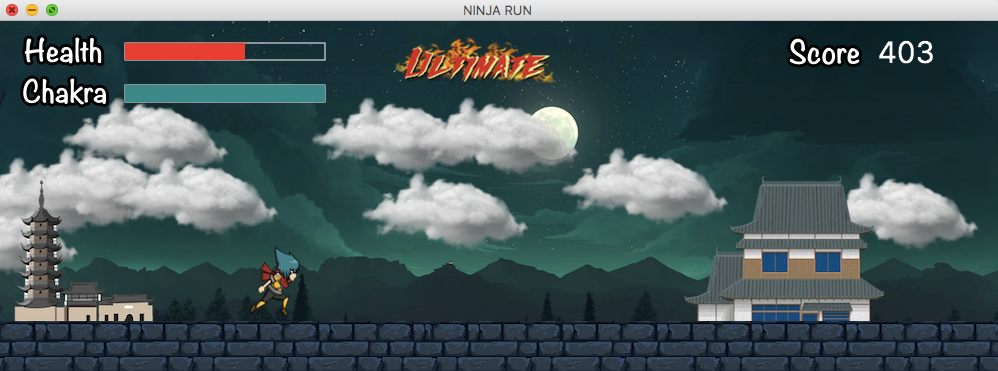


Figure 4: Ultimate skill alert

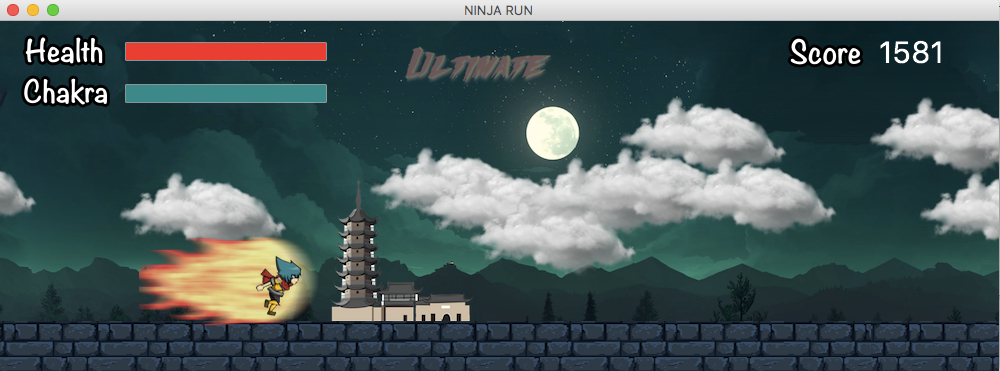


Figure 5: Use ultimate skill

At play game state, player can.

* Press UP to jump
* Press RIGHT to warp and Chakra gauge is cool down
* Press LEFT to use ultimate skill when ultimate skill alert
* Press DOWN to sprint
* Press SPACE BAR to attack
* Press ESC to Pause

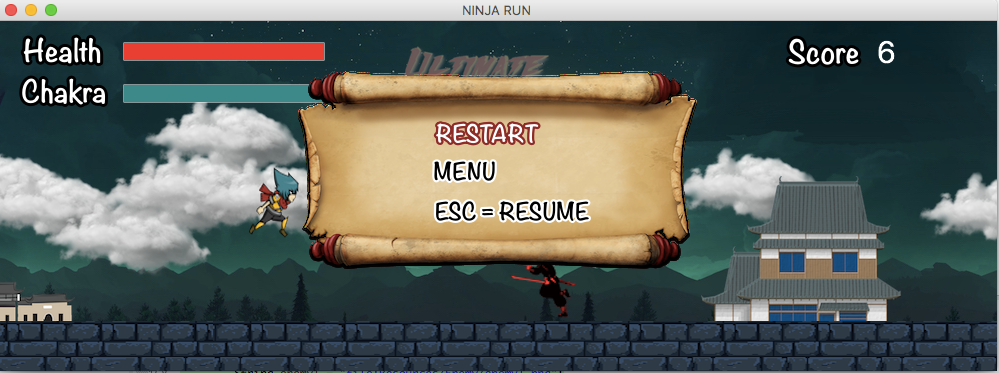
When player press ESC , game is paused and display pause state.

Figure 6: the Pause state

At Pause state, player can

* Press ESC to resume
* Press Enter to select option.
* Press Up/Down arrow to move option.

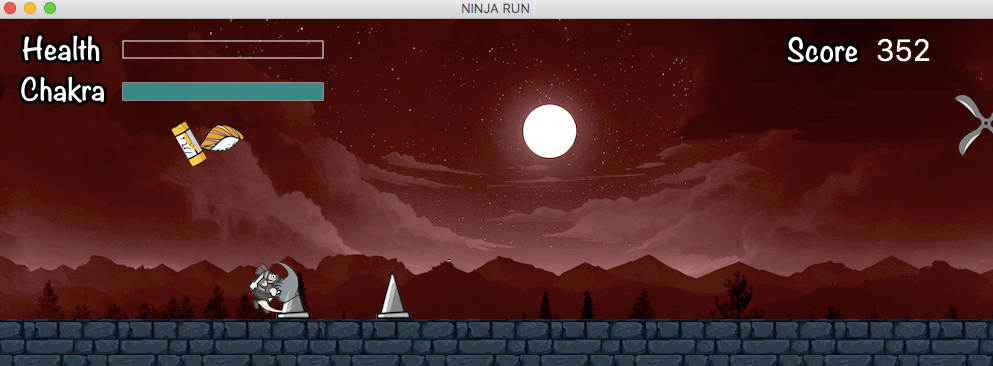
When health gauge is run out , ninja is died. The screen switch to game over state.

Figure 7: When ninja die

-Press Enter to go to game over state

This state will show your score and press enter to return menu.

Figure 7 : Game over state

At game over state, player can

* Press Enter to return menu

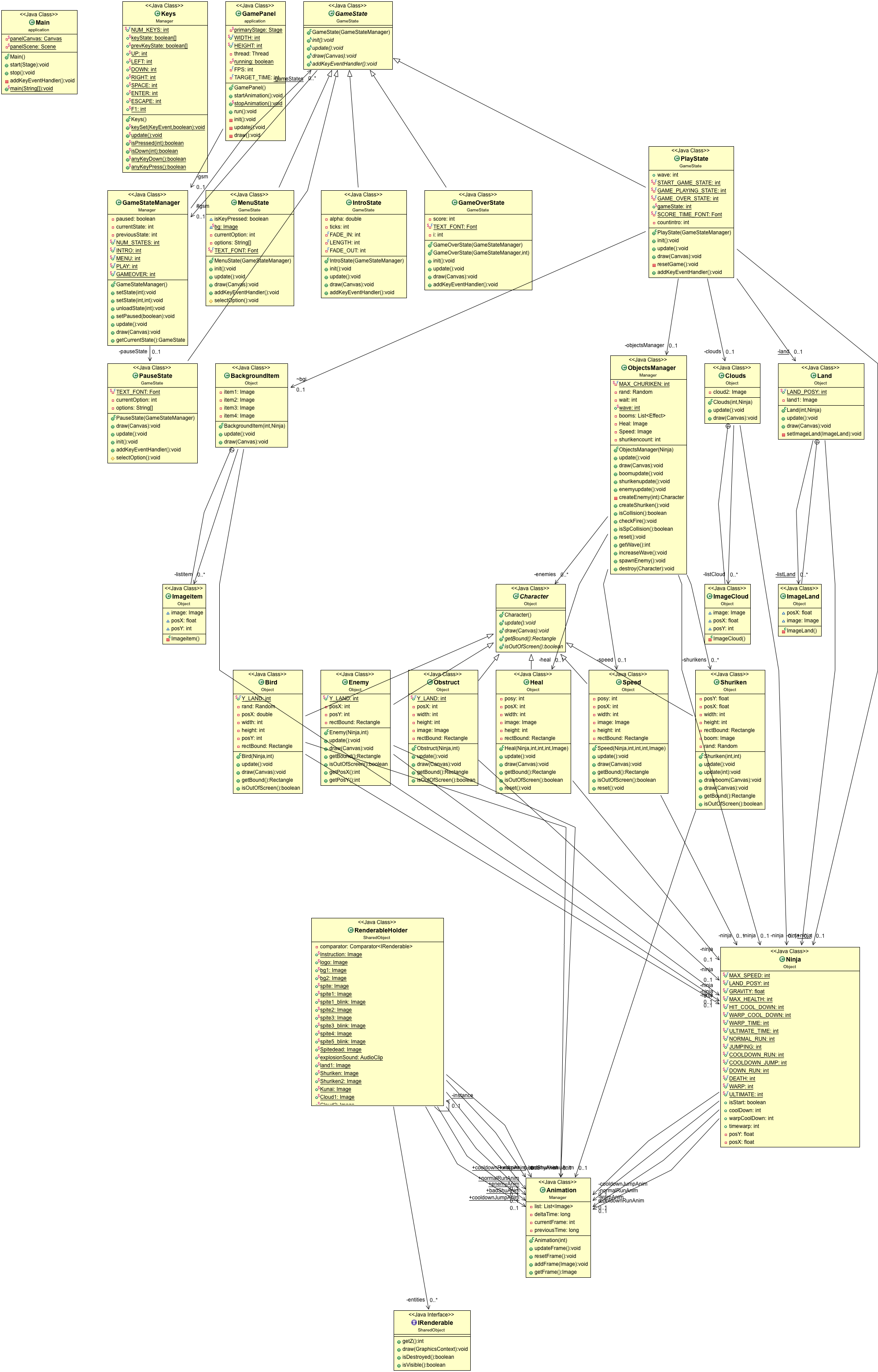
2.Implementation Detail

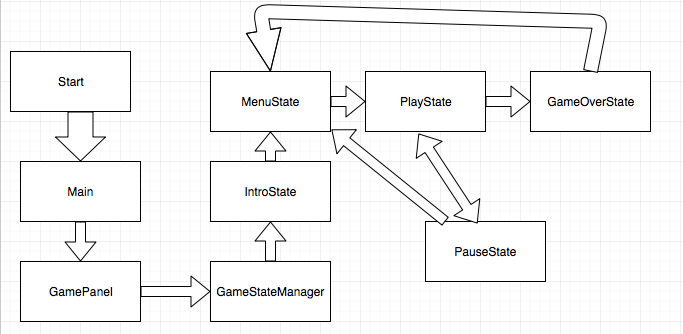
Figure 8 : UML diagram of the program.

Figure 9 : The Relation of State class.

When the application starts ,Main call GamePanel and GamePanel call GameStateManager to draw the IntroState. So next state is MenuState ,It can stat game and exit game.When game is started PlayState draw a game , If we press ESC , It game pauses and shows pause screen. These can return to MenuState. When we lost game, State changes to GameOverState . It shows score and return MenuState

|  |  |
| --- | --- |
| -Canvas panelCanvas | initialize GamePanel. |
| -Scene panelScene | initialize scene by GamePanal |

EveryState was control by GamePanel to update it every second and GameStateManager to switch state.

|  |  |
| --- | --- |
| -static Stage primaryStage | a primaryStage |
| + static final int WIDTH | set width = 1000 |
| + static final int HEIGHT | set height = 350; |
| -Thread thread | A thread for game animation. |
| -static boolean running | Is game running |
| - int FPS | Number of frame rates per second. Default is 30. |
| -int TARGET\_TIME | Time period between each update of a game animation ,Default is 1000/FPS |
| -GameStateManager | A game state manager. |

2.1 Package application

2.1.1 Class Main extends Application

2.1.1.1 Field

2.1.1.2 Method

|  |  |
| --- | --- |
| +void start(Stage primaryStage) | The main entry point for the JavaFX applications. |
| +void stop() | This method are going to execute before JavaFX application terminates. In this program, we use it to stop the remaining threads before exit the program. |
| + void main(String[] args) | An entry point of the application. |
| -void addKeyEventHandler() | Add two KeyEvent handlers to the canvas.  - When player press any letter key (A-Z) (KeyPressed), use characterInput to add pressed character.  - When player release any letter key (A-Z) (KeyReleased),  use CharacterInput to remove pressed character. |

2.1.2 Class GamePanel extends Canvas implements Runnable

2.1.2.1 Field

2.1.2.2 Constructor

|  |  |
| --- | --- |
| +GamePanel() | Initialize GamePanel,Set width = WIDTH ,  height =HEIGHT  -requestFocus()  -startAnimation() |

2.1.2.3 Method

|  |  |
| --- | --- |
| +void startAnimation() | Start the game animation loop thread and set Running to true. |
| +void stopAnimation() | set Running to false. |
| + void run() | A game animation loop.  -The loop will stop when Running is false.  -The animation will update when the time has pass by a TARGET\_TIME  -For each loop, call update() and draw(). |
| -void init() | Initialize GameStateManager.  set Running to true |
| -void update | GameStateManager and Keys update |
| -void draw | GameStateManager draw in GamePanel. |